

# Guía de Visualino

Víctor R. Ruiz <rvr@linotipo.es>  
6 de abril de 2015

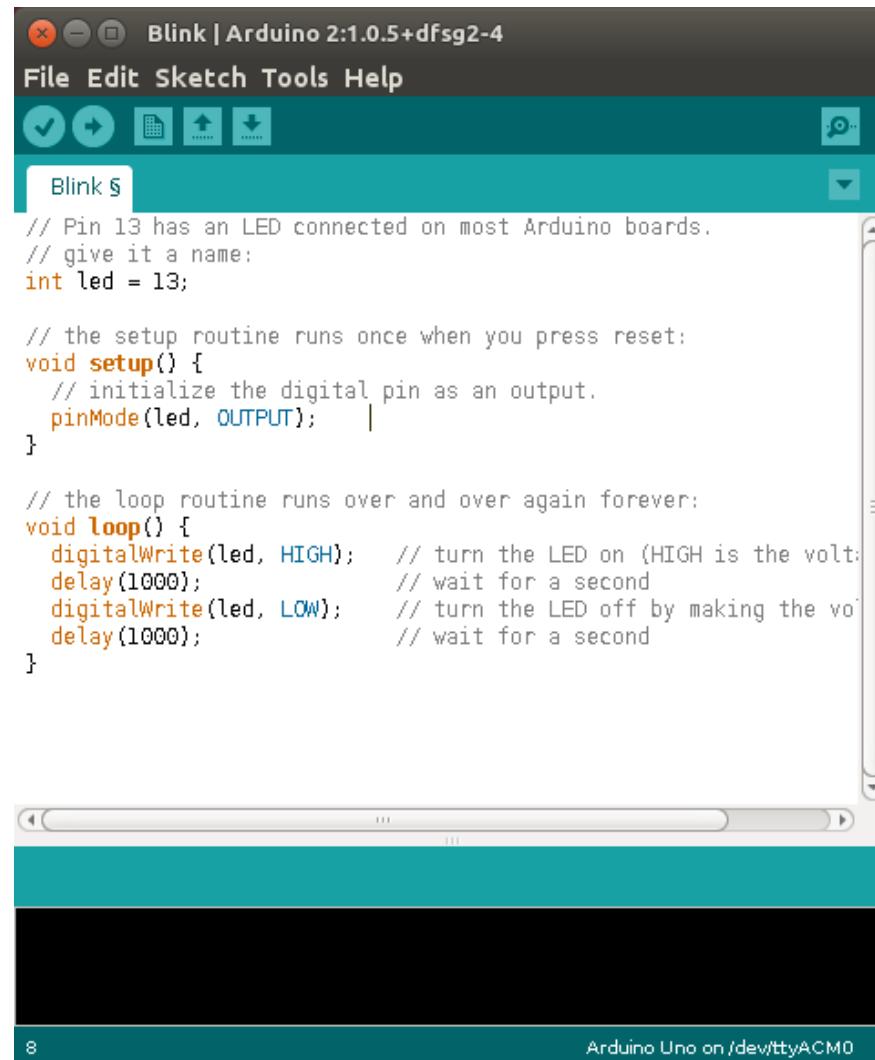


# Índice

1. Entorno.
2. Programación visual.
3. Prácticas.
4. Recursos.

# Entorno

# IDE de Arduino



The image shows a screenshot of the Arduino IDE. The title bar reads "Blink | Arduino 2:1.0.5+dfsg2-4". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The toolbar has icons for save, undo, redo, and upload. The text editor contains the "Blink" sketch, which is a classic example of an LED blinking program. The code is as follows:

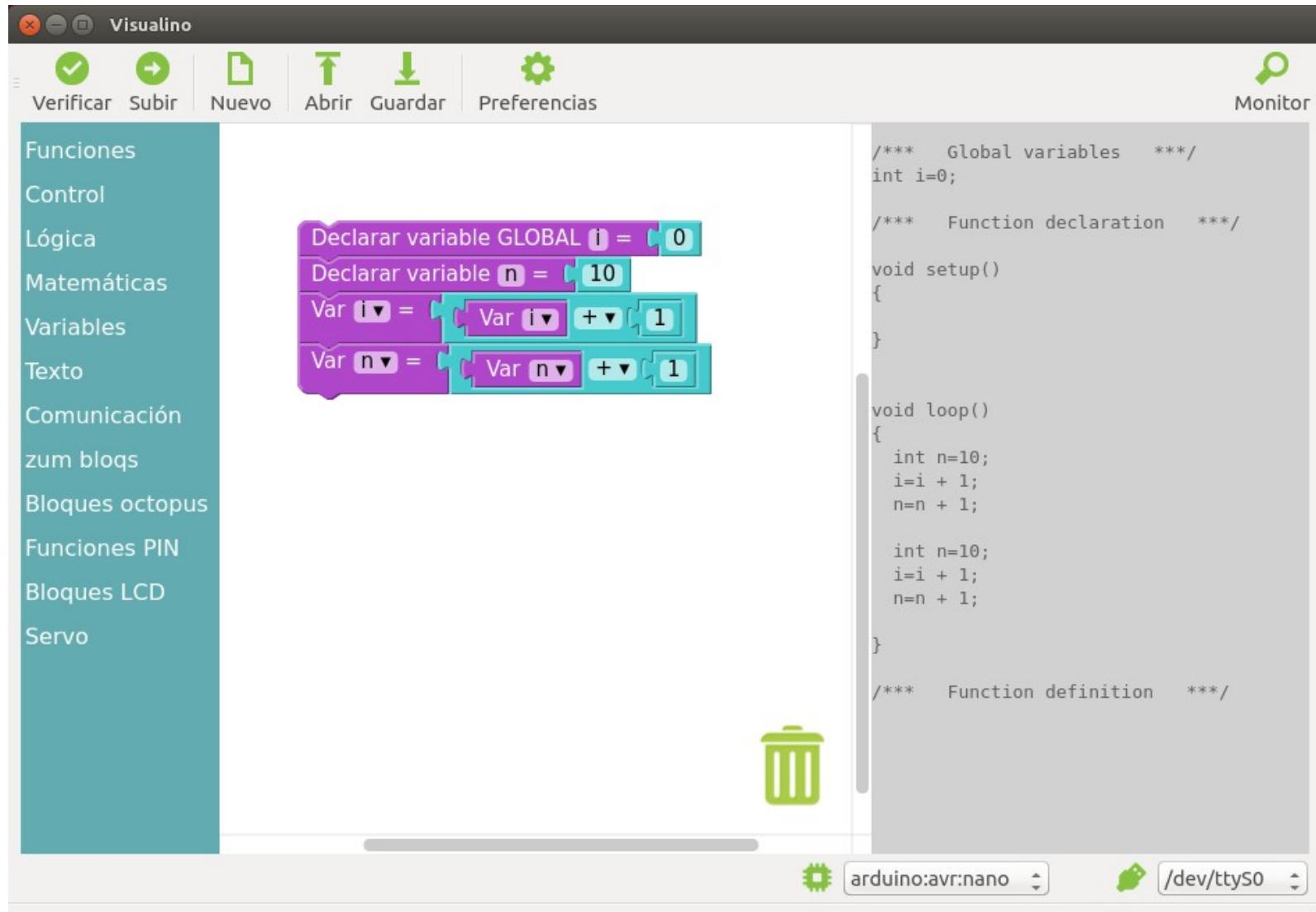
```
// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

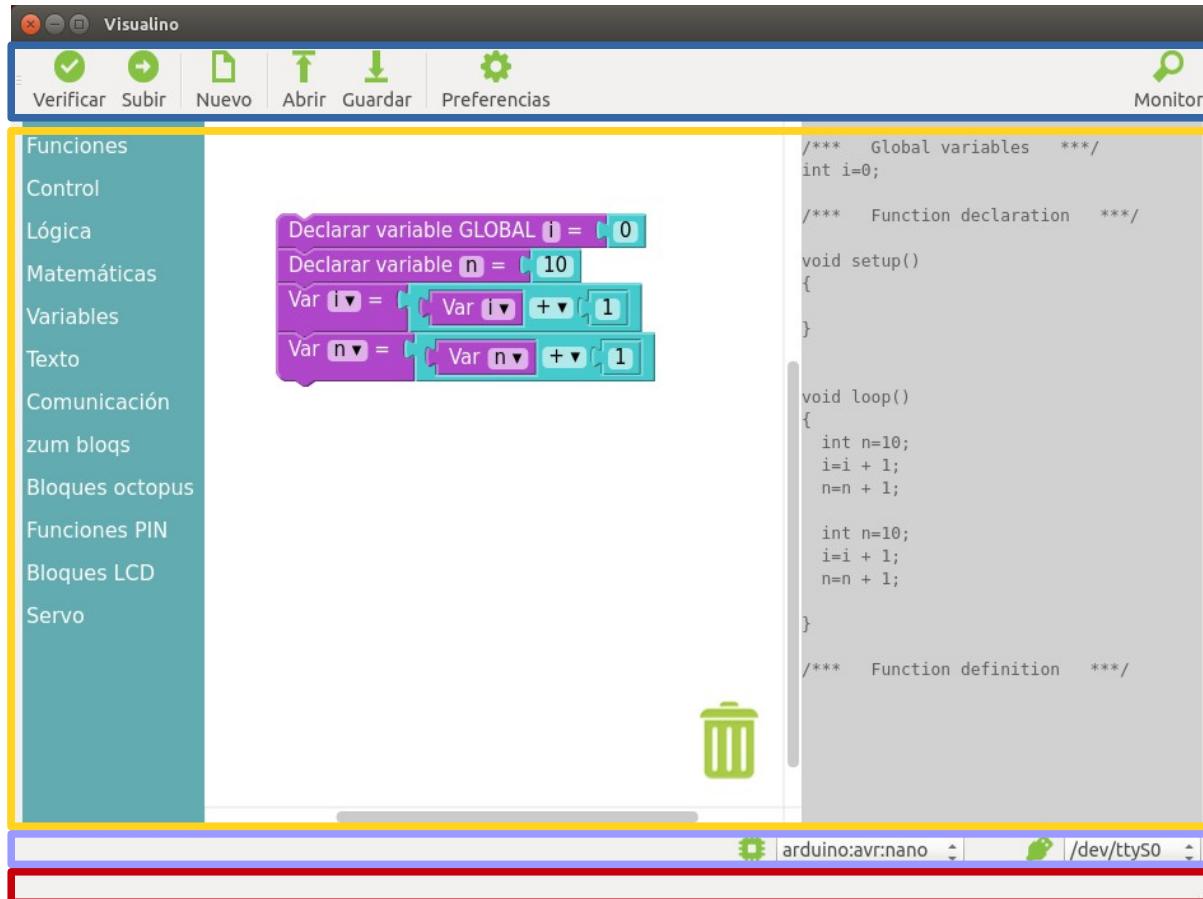
// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);    // turn the LED on (HIGH is the volt:
  delay(1000);               // wait for a second
  digitalWrite(led, LOW);     // turn the LED off by making the vo:
  delay(1000);               // wait for a second
}
```

The status bar at the bottom shows "8" and "Arduino Uno on /dev/ttyACM0".

# IDE de Visualino



# IDE de Visualino



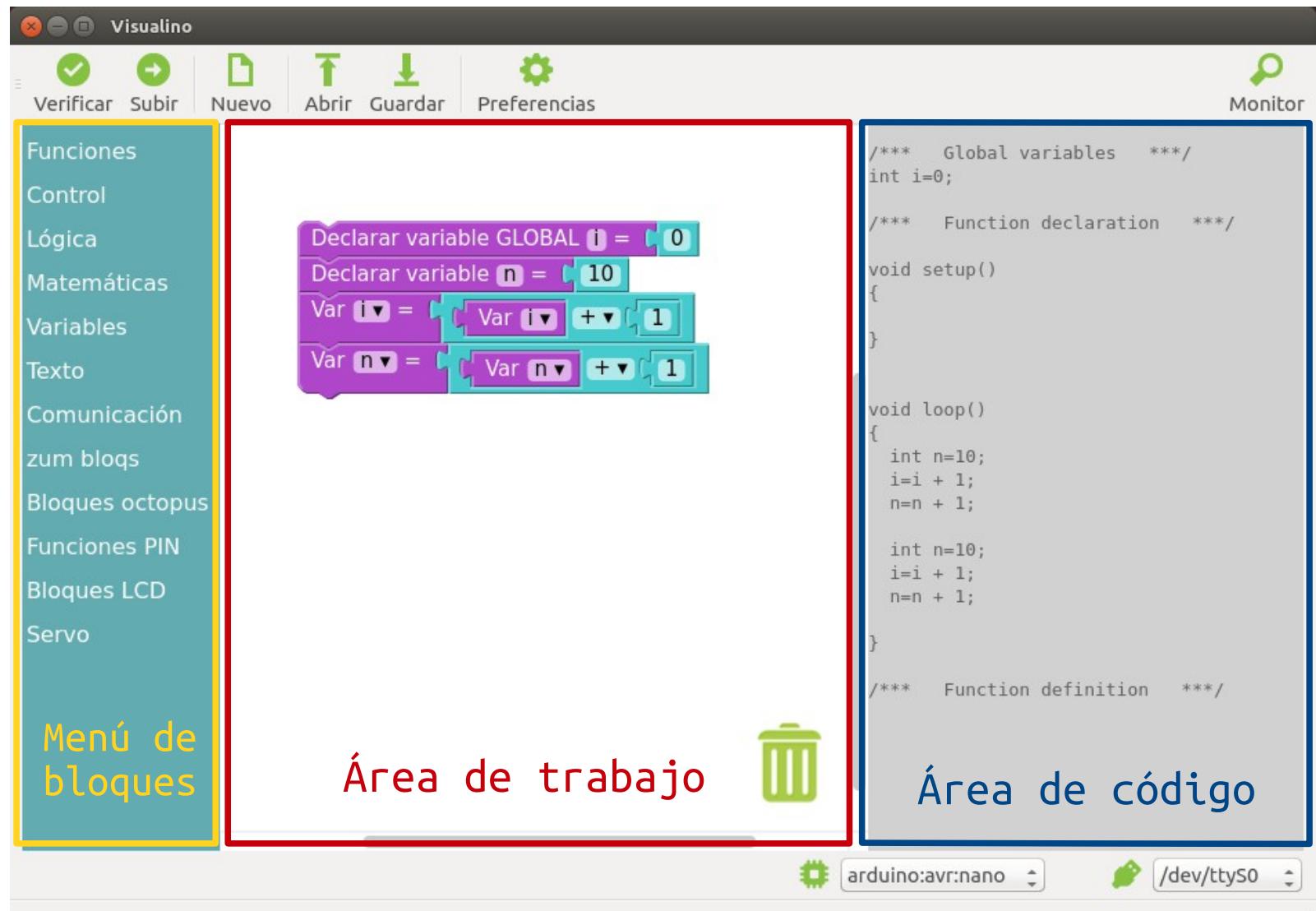
Barra de iconos

Editor

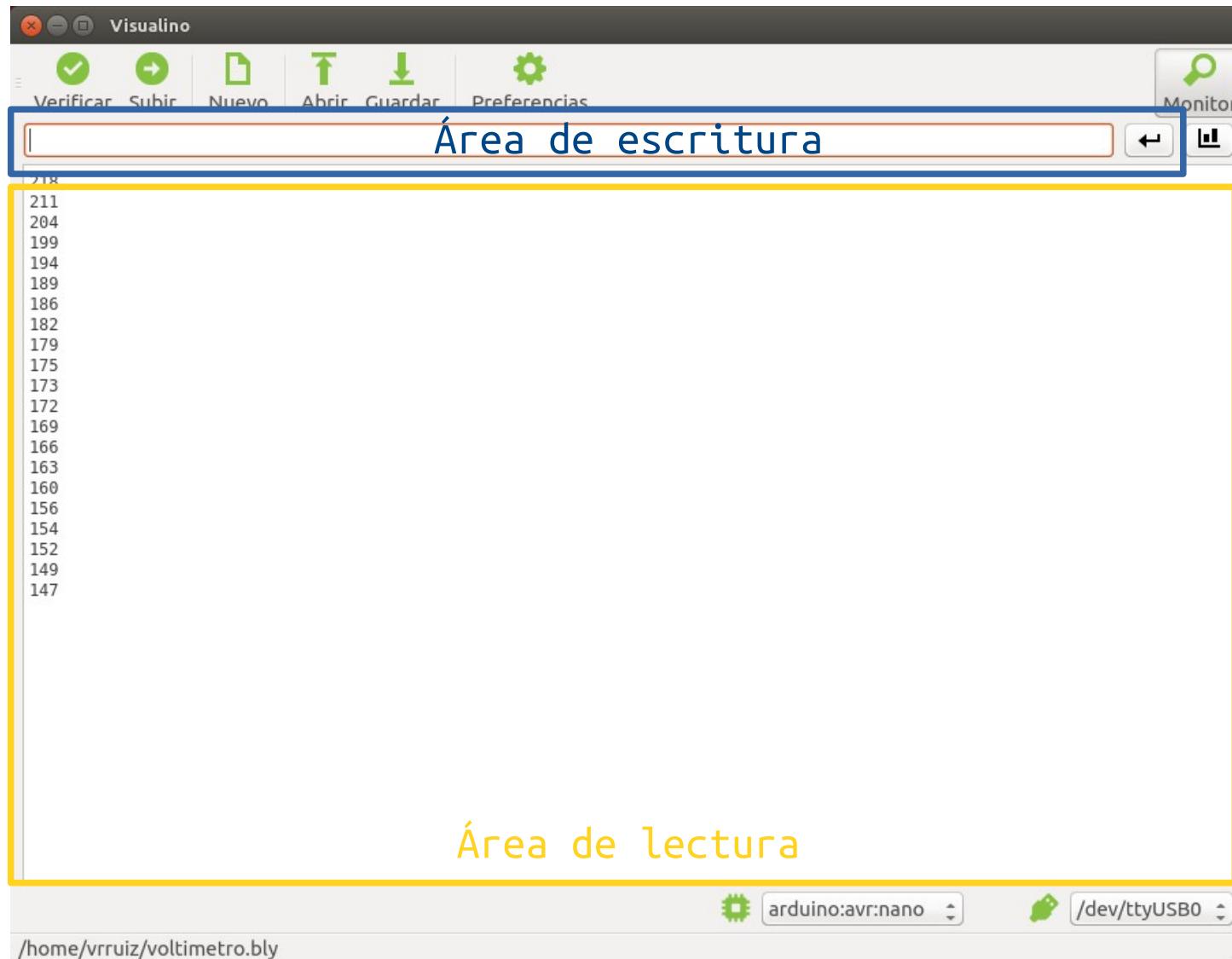
Barra de comunicaciones

Barra de estado

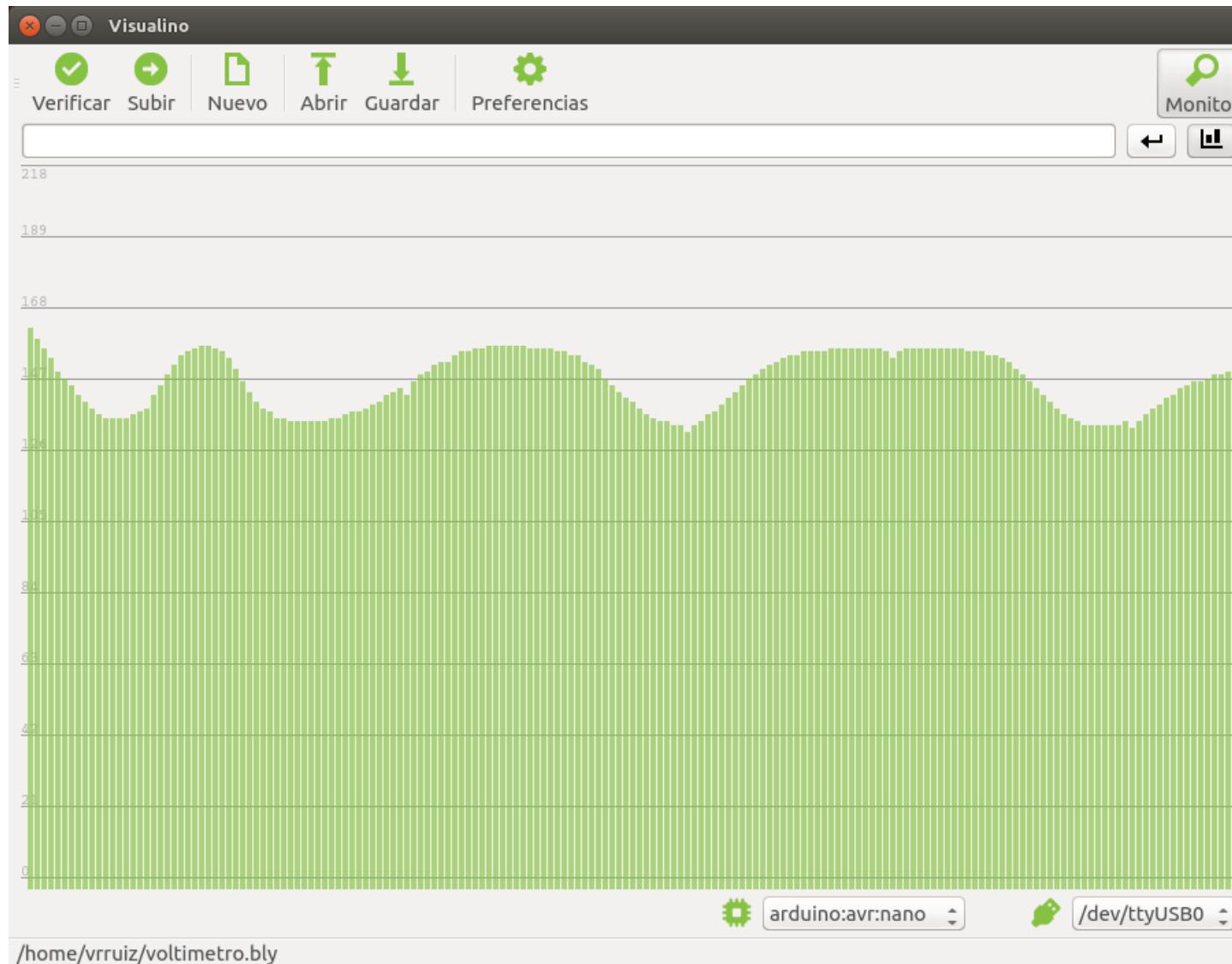
# Editor



# Monitor serie



# Monitor: Gráficos



# Programación visual

# Operativa

1. Crear programa.
2. Verificar programa.
3. Subir programa.
4. Monitor serie (opcional).

# Bloques principales

- Control.
- Lógica.
- Matemáticas.
- Texto.
- Comunicación.
- Funciones PIN.

Funciones  
Control  
Lógica  
Matemáticas  
Variables  
Texto  
Comunicación  
zum bloqs  
Bloques octopus  
Funciones PIN  
Bloques LCD  
Servo

# Tipos de bloques

- Instrucción.
- Asignación.
- Valor.
- Funciones.

# Bloques: Instrucciones



Escribir en PIN digital el valor analógico

Imprimir por puerto serie con salto de línea

Esperar [ms]

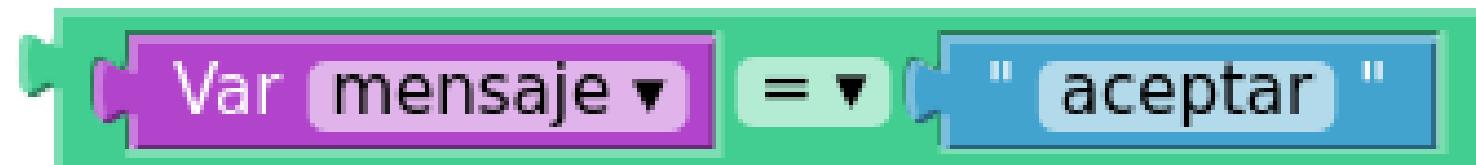
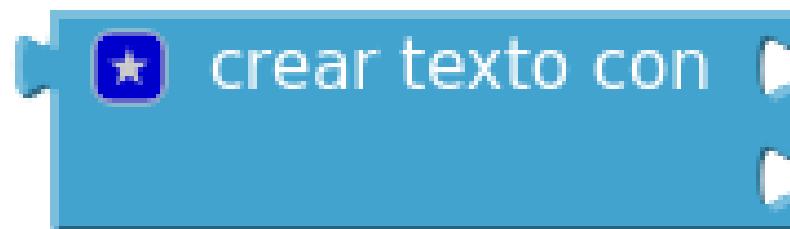
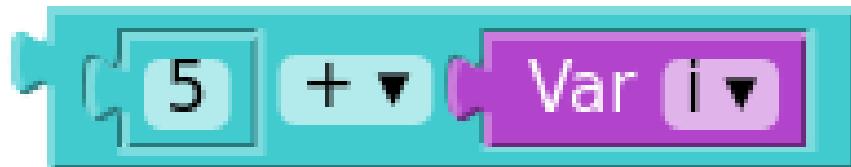
a añadirle texto

# Bloques: Asignaciones

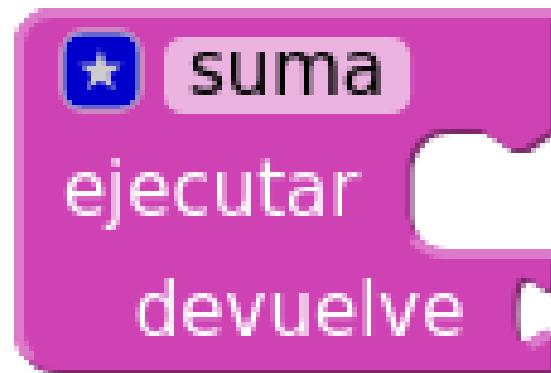
Declarar variable GLOBAL  = 

Declarar variable  = 

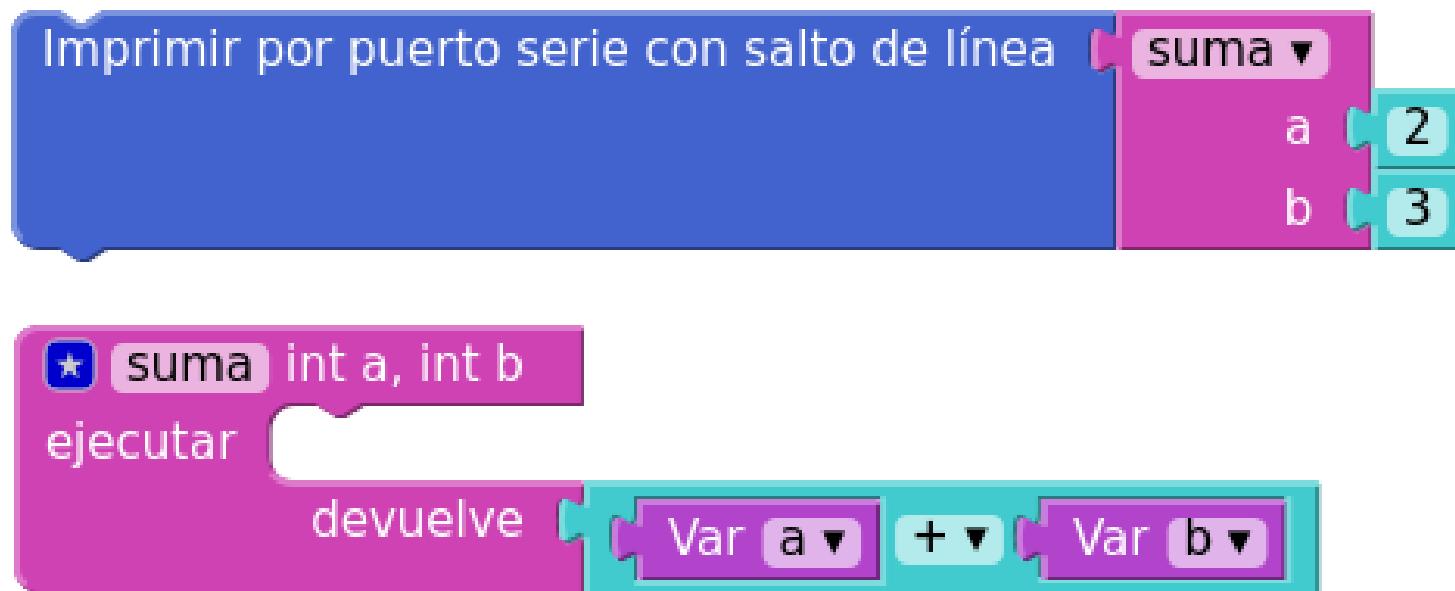
# Bloques: Valores



# Bloques: Funciones

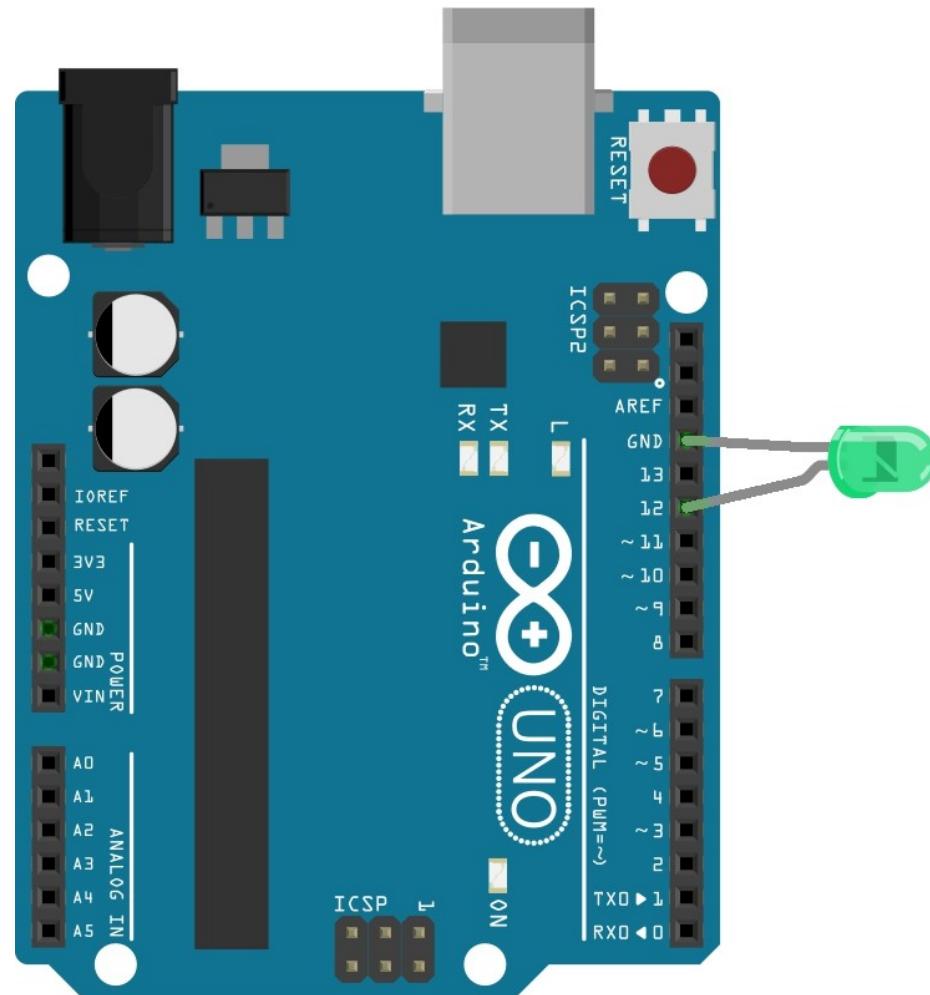


# Encajar las piezas del puzzle



# Prácticas

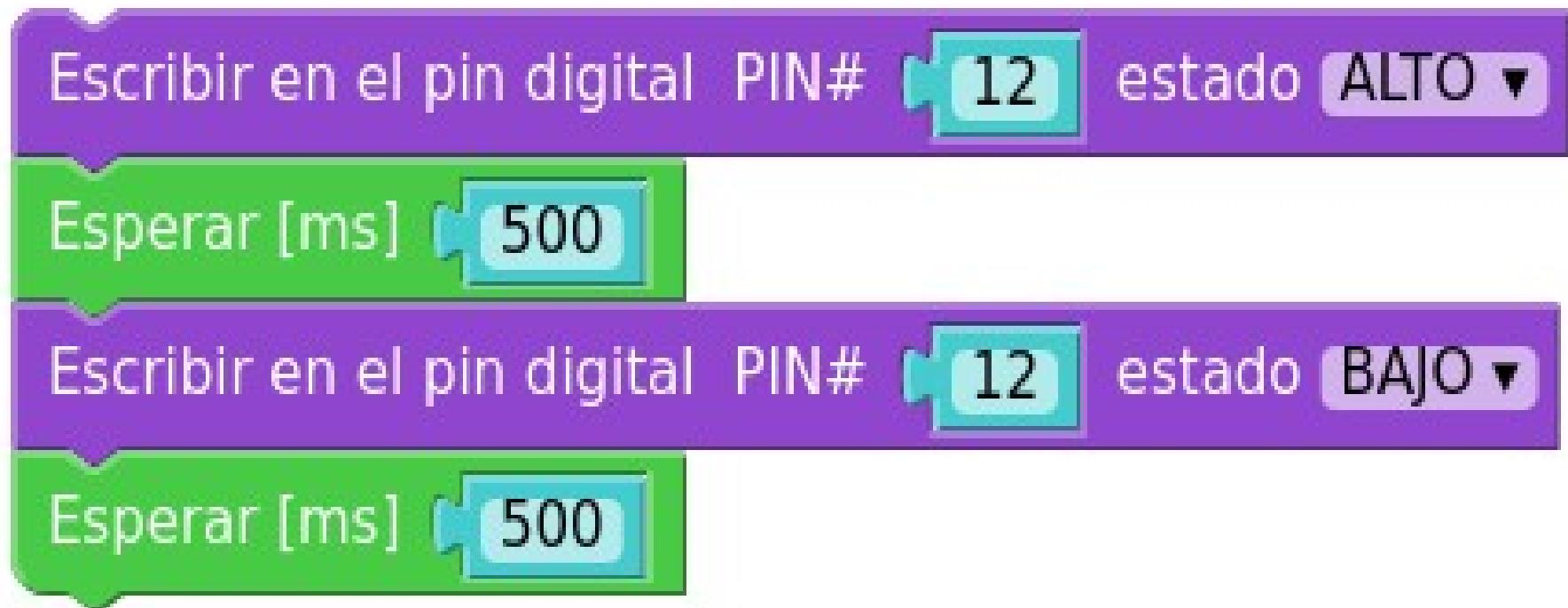
# Parpadeo LED en pin digital



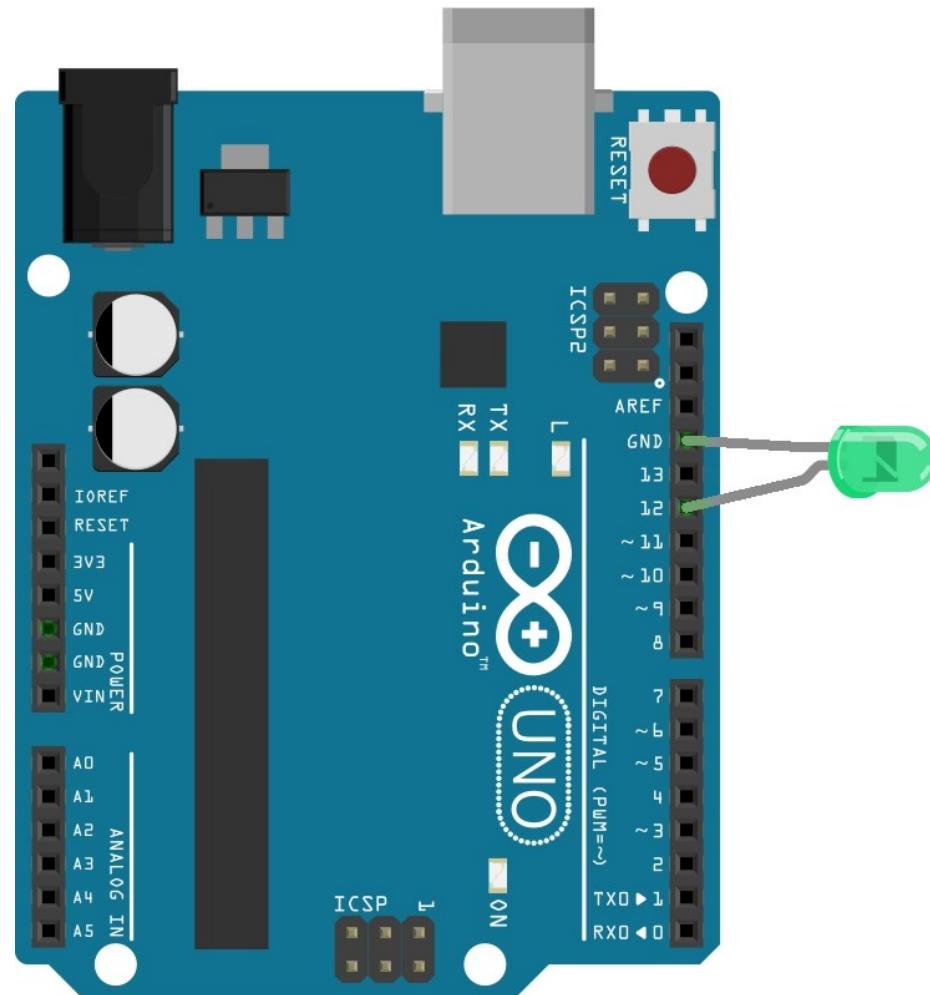
fritzing

Visual ino  
powered by Roboblocks

# Parpadeo LED en pin digital



# Parpadear un LED 3 veces



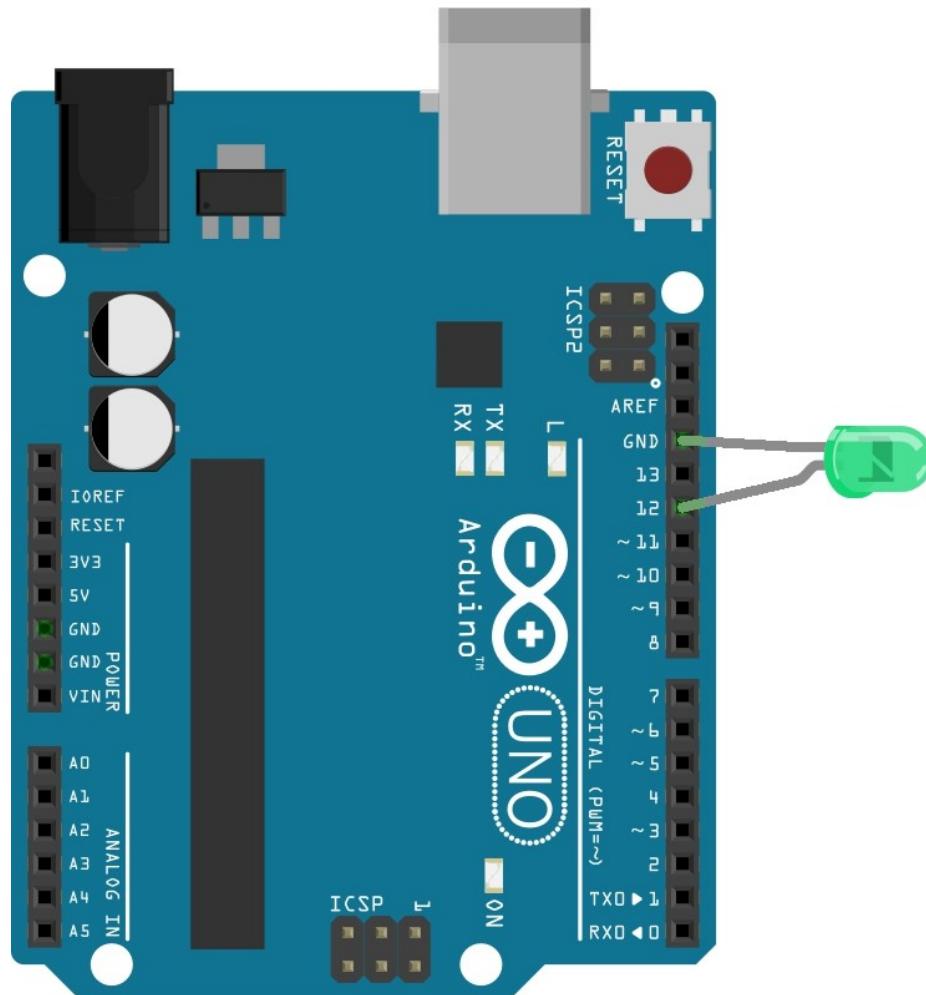
fritzing

Visual ino  
powered by Roboblocks

# Parpadear un LED 3 veces



# Parpadear un LED 3 veces repetitiva



fritzing

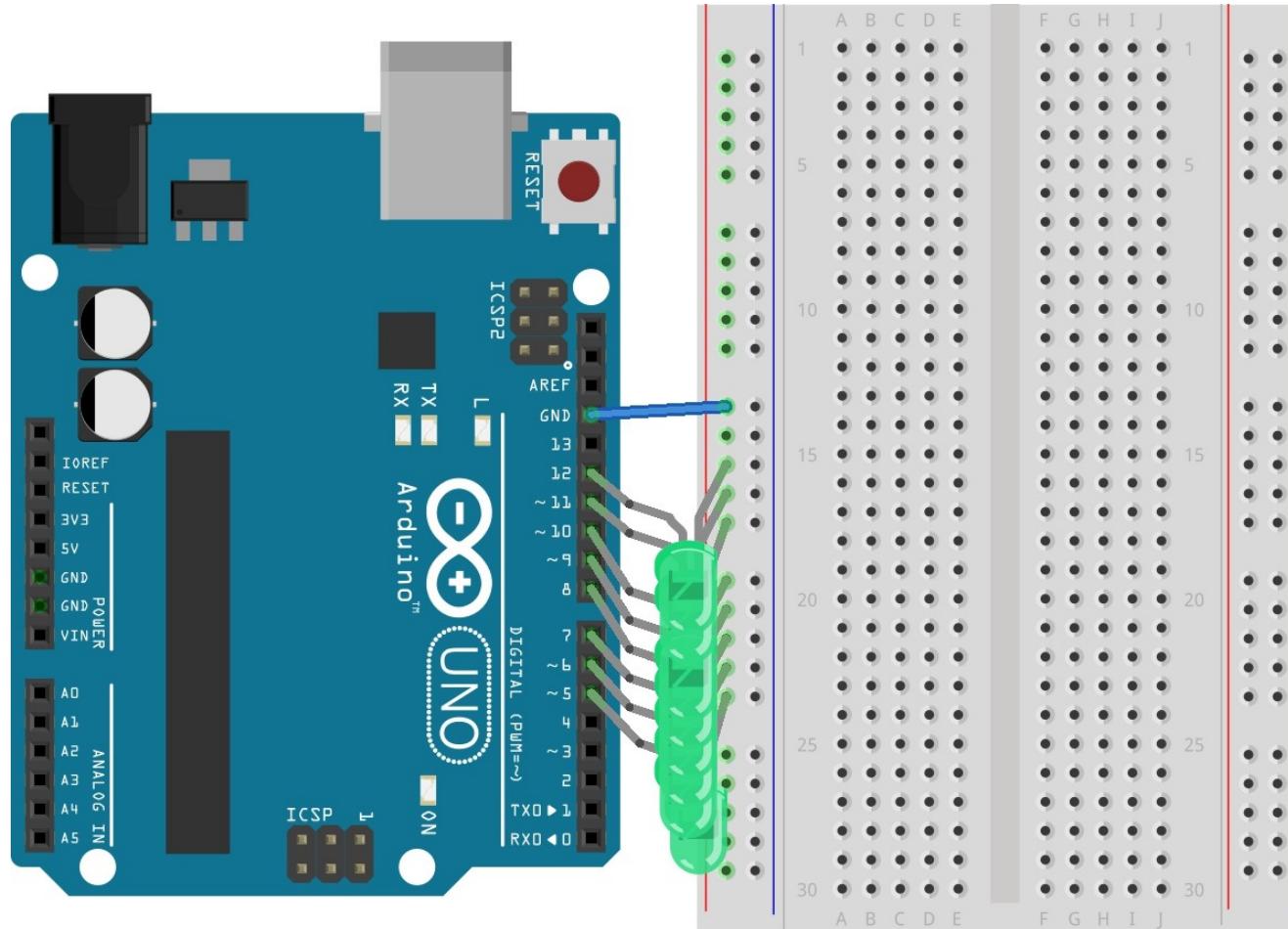
Visual-ino  
powered by Roboblocks

# Parpadear un LED 3 veces repetitiva

```
Declarar variable GLOBAL i = 0
Declarar variable GLOBAL once = falso
si Var once = falso
  ejecutar Contar con Var i desde 0 hasta 4
    ejecutar
      Escribir en el pin digital PIN# 12 estado ALTO
      Esperar [ms] 500
      Escribir en el pin digital PIN# 12 estado BAJO
      Esperar [ms] 500
  Var once = verdadero
```

The image shows a Scratch script for a microcontroller. It starts with two global variable declarations: 'i' set to 0 and 'once' set to 'falso' (false). A 'si' (if) block checks the value of 'once'. If it is 'falso' (false), it enters a 'ejecutar' (execute) loop. Inside this loop, it runs a 'Contar con' (count with) loop from 0 to 4. Within this, it runs three 'Escribir en el pin digital' (write to digital pin) blocks: one for pin 12 at 'ALTO' (high), one for pin 12 at 'BAJO' (low), and another for pin 12 at 'ALTO' (high). After the inner loop, it sets 'once' to 'verdadero' (true).

# Coche fantástico (opcional)

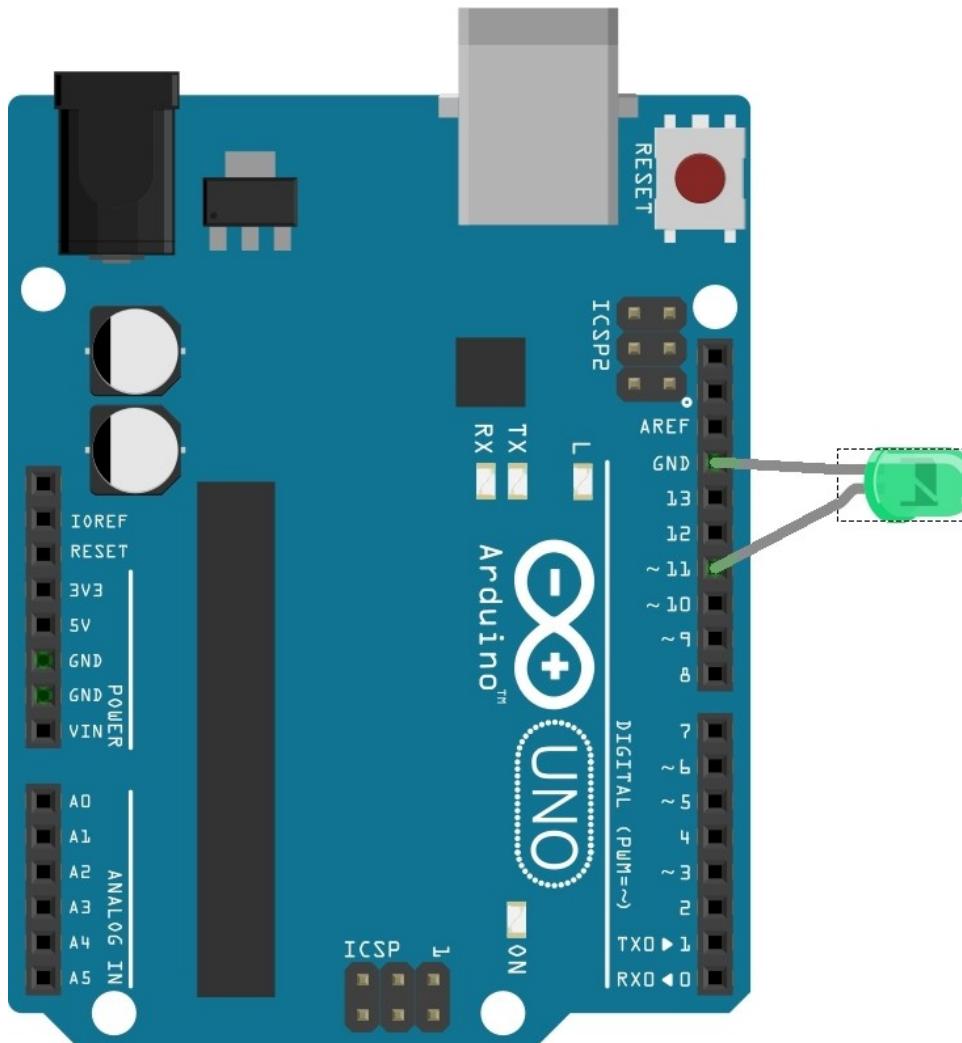


fritzing

# Coche fantástico (opcional)

```
Declare variable pin equals 0
count with Var pin from 5 to 13
do
  digitalWrite PIN# Var pin state HIGH
  Wait (ms) 200
count with Var pin from 13 to 5
do
  digitalWrite PIN# Var pin state HIGH
  Wait (ms) 200
```

# Fluctuación continua de brillo



# Fluctuación continua de brillo

```
Declarar variable brigthness = 0
Contar con Var brigthness ▾ desde 0 hasta 255
ejecutar
  Escribir en PIN digital 11 el valor analógico Var brigthness ▾
  Esperar [ms] 200
Contar con Var brigthness ▾ desde 255 hasta 0
ejecutar
  Escribir en PIN digital 11 el valor analógico Var brigthness ▾
  Esperar [ms] 200
```

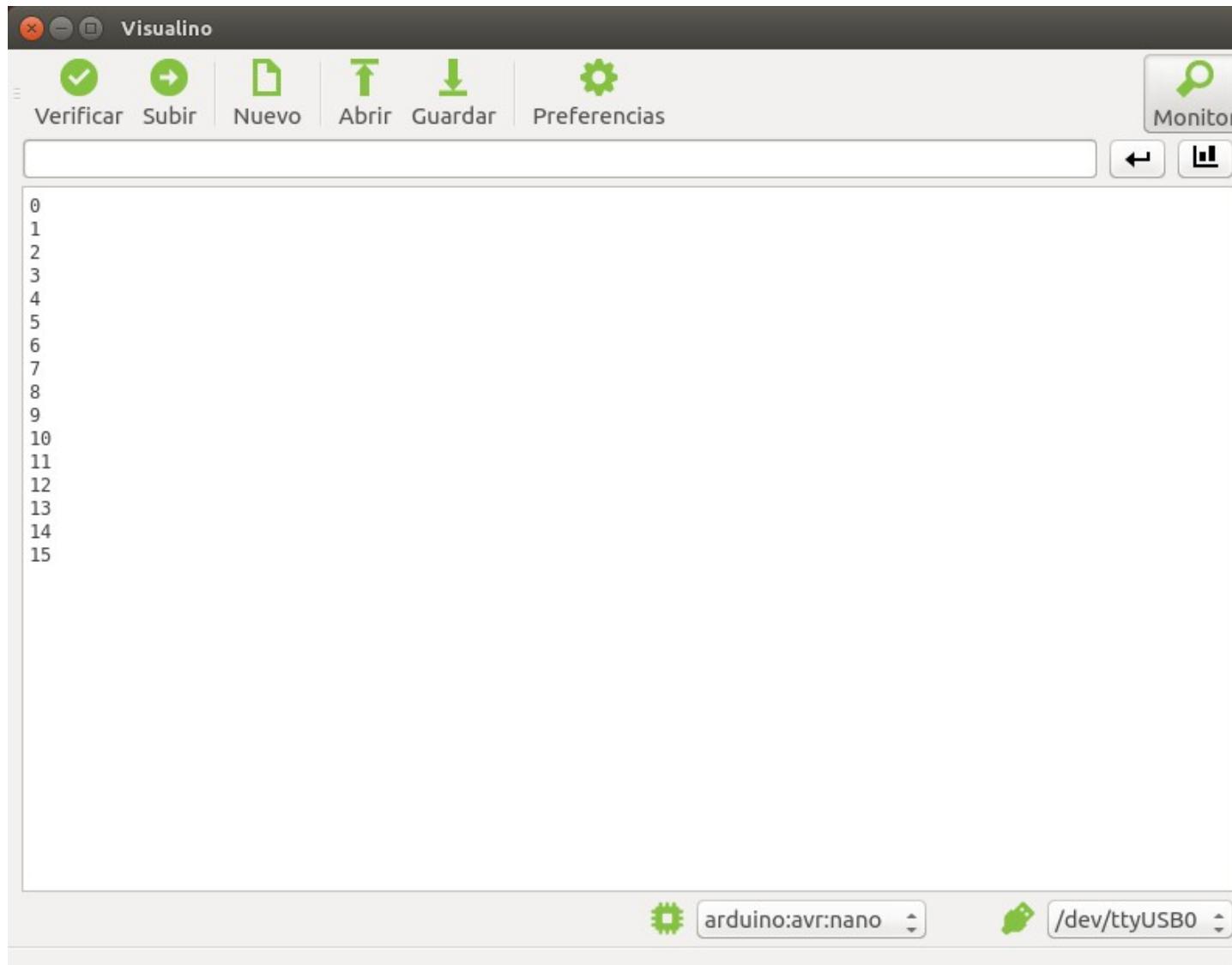
# Contador



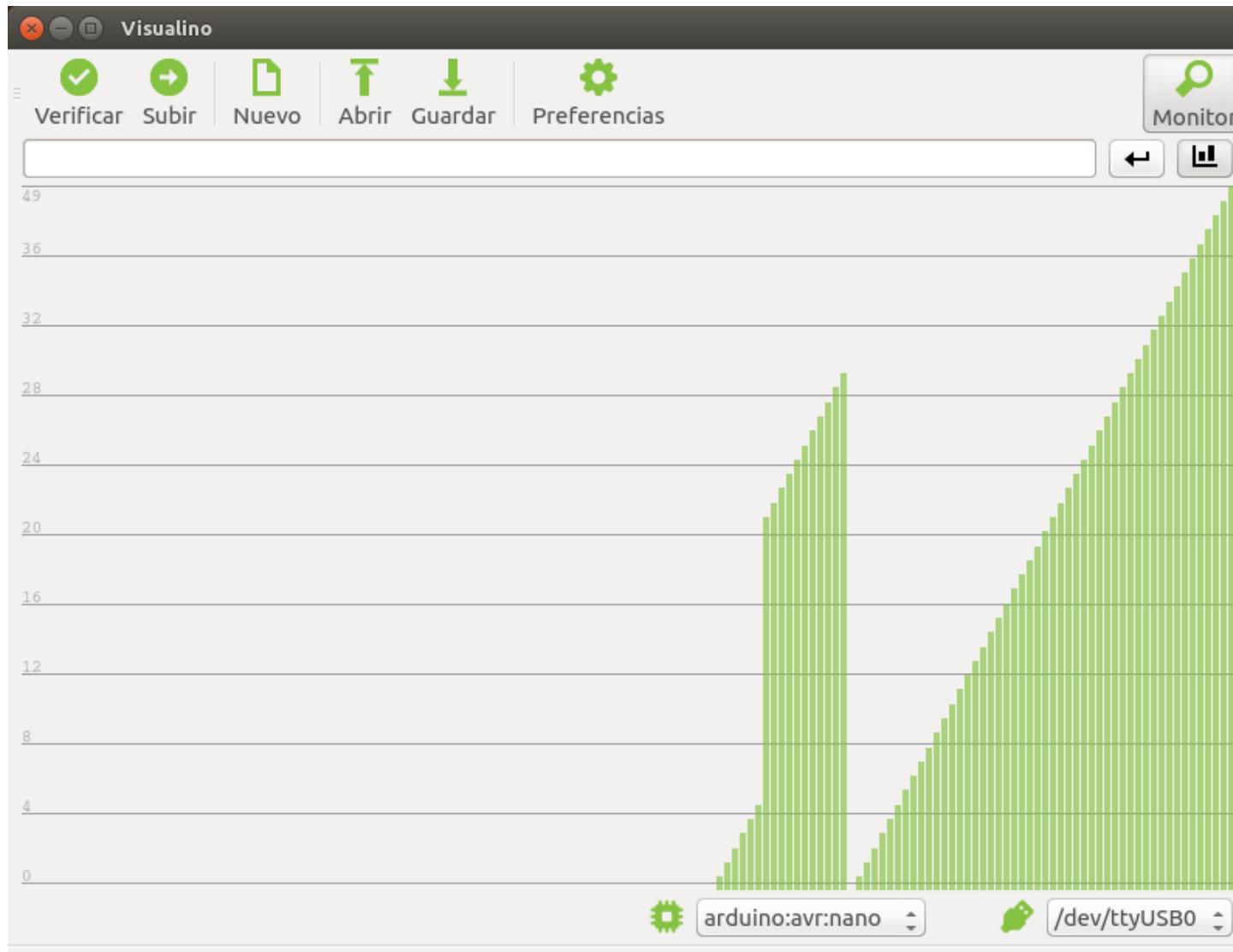
Scratch script:

- Declarar variable GLOBAL **i** = 0
- Imprimir por puerto serie con salto de línea | Var **i** |
- Var **i** = | Var **i** + 1 |
- Esperar [ms] | 1000 |

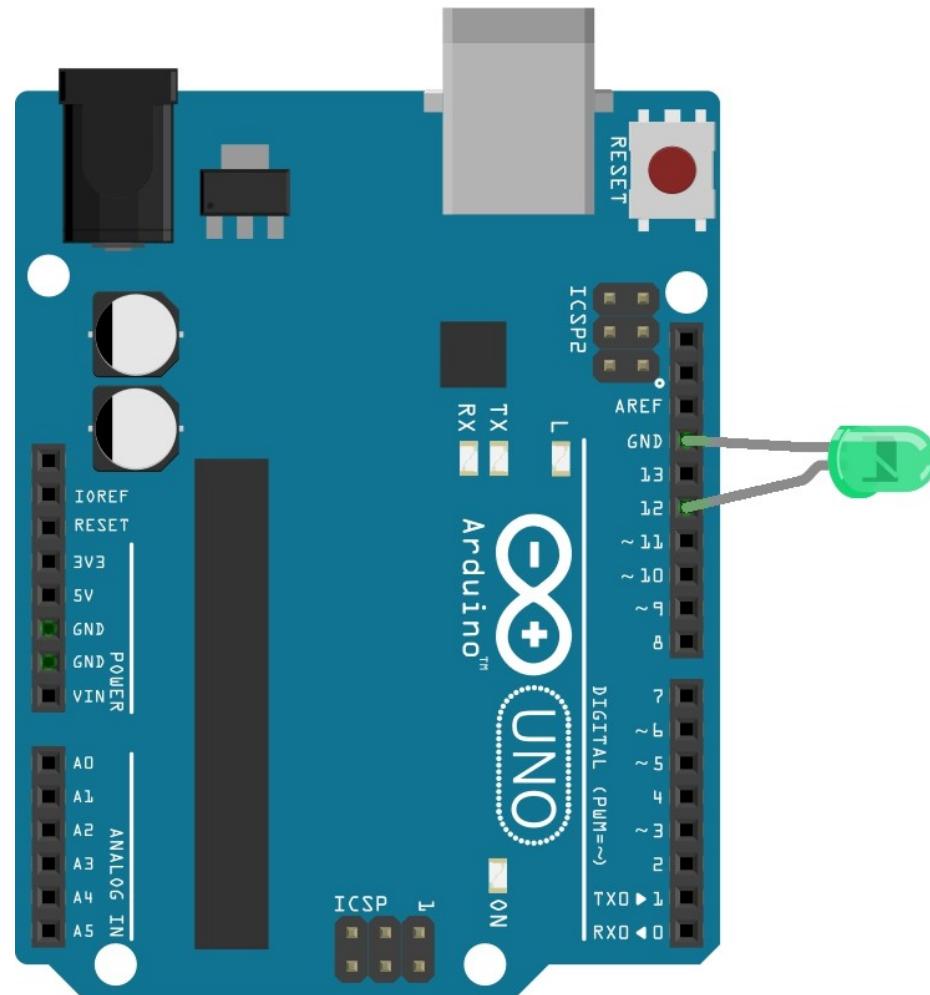
# Contador: Monitor



# Contador: Gráficas



# Interruptor



fritzing

Visual  + ino  
powered by Roboblocks

# Interruptor

```
Declarar variable message = 0
Puerto Serie Disponible
ejecutar Var message = Leer desde el puerto serie
Imprimir por puerto serie con salto de línea Var message
  si Var message = 48
    ejecutar Escribir en el pin digital PIN# 12 estado BAJO
  si Var message = 49
    ejecutar Escribir en el pin digital PIN# 12 estado ALTO
```

# Recursos

# Recursos

- Documentación de Visualino.
- Tutoriales de Bitbloq (bq).
- Documentación de Arduino.
- Documentación de Arduino Gran Canaria.

# [www.visualino.net](http://www.visualino.net)

Descargar: [visualino.net/downloads/](http://visualino.net/downloads/)

Documentación: [visualino.net/docs/](http://visualino.net/docs/)

